

Source Separation using Regularized NMF with MMSE Estimates under GMM Priors with Online Learning for The Uncertainties

Emad M. Grais and Hakan Erdogan
 {grais, haerdogan}@sabanciuniv.edu

*Faculty of Engineering and Natural Sciences,
 Sabanci University, Orhanli Tuzla, 34956, Istanbul.*

Abstract

We propose a new method to enforce priors on the solution of the nonnegative matrix factorization (NMF). The proposed algorithm can be used for denoising or single-channel source separation (SCSS) applications. The NMF solution is guided to follow the Minimum Mean Square Error (MMSE) estimates under Gaussian mixture prior models (GMM) for the source signal. In SCSS applications, the spectra of the observed mixed signal are decomposed as a weighted linear combination of trained basis vectors for each source using NMF. In this work, the NMF decomposition weight matrices are treated as a distorted image by a distortion operator, which is learned directly from the observed signals. The MMSE estimate of the weights matrix under GMM prior and log-normal distribution for the distortion is then found to improve the NMF decomposition results. The MMSE estimate is embedded within the optimization objective to form a novel regularized NMF cost function. The corresponding update rules for the new objectives are derived in this paper. Experimental results show that, the proposed regularized NMF al-

gorithm improves the source separation performance compared with using NMF without prior or with other prior models.

Keywords: Single channel source separation, nonnegative matrix factorization, minimum mean square error estimates, and Gaussian mixture models.

1. Introduction

Nonnegative matrix factorization (Lee and Seung, 2001) is an important tool for source separation applications, especially when only one observation of the mixed signal is available. NMF is used to decompose a nonnegative matrix into a multiplication of two nonnegative matrices, a basis matrix and a gains matrix. The basis matrix contains a set of basis vectors and the gains matrix contains the weights corresponding to the basis vectors in the basis matrix. The NMF solutions are found by solving an optimization problem based on minimizing a predefined cost function. As most optimization problems, the main goal in NMF is to find the solutions that minimize the cost function without considering any prior information rather than the nonnegativity constraint. There have been many works that tried to enforce prior information related to the nature of the application on the NMF decomposition results. For audio source separation applications, the continuity and sparsity priors were enforced in the NMF decomposition weights (Virtanen, 2007). In Bertin et al. (2009), and Bertin et al. (2010), smoothness and harmonicity priors were enforced on the NMF solution in Bayesian framework and applied to music transcription. In Wilson et al. (2008b), and Wilson et al. (2008a) the regularized NMF was used to increase the NMF decomposition

weights matrix likelihood under a prior Gaussian distribution. In Fevotte et al. (2009), Markov chain prior model for smoothness was used within a Bayesian framework in regularized NMF with Itakura-Saito (IS-NMF) divergence. In Virtanen et al. (2008), the conjugate prior distributions on the NMF weights and basis matrices solutions with the Poisson observation model within Bayesian framework was introduced. The Gamma distribution and the Gamma Markov chain (Cemgil and Dikmen, 2007) were used as priors for the basis and weights/gains matrices respectively in Virtanen et al. (2008). A mixture of Gamma prior model was used as a prior for the basis matrix in Virtanen and Cemgil (2009). The regularized NMF with smoothness and spatial decorrelation constraints was used in Chen et al. (2006) for EEG applications. In Cichocki et al. (2006), and Chen et al. (2006), a variety of constrained NMF algorithms were used for different applications.

In supervised single channel source separation (SCSS), NMF is used in two main stages, the training stage and the separation stage (Schmidt and Olsson, 2006; Grais and Erdogan, 2011a,b,c; Grais et al., 2012; Grais and Erdogan, 2012c). In the training stage, NMF is used to decompose the spectrogram of clean training data for the source signals into a multiplication of trained basis and weights/gains matrices for each source. The trained basis matrix is used as a representative model for the training data of each source and the trained gains matrices are usually ignored. In the separation stage, NMF is used to decompose the mixed signal spectrogram as a nonnegative weighted linear combination of the columns in the trained basis matrices. The spectrogram estimate for each source in the mixed signal can be found by summing its corresponding trained basis terms from the NMF decomposition during the

separation stage. One of the main problems of this framework is that, the estimate for each source spectrogram is affected by the other sources in the mixed signal. The NMF decomposition of the weight combinations in the separation stage needs to be improved. To improve the NMF decomposition during the separation stage, prior information about the weight combinations for each source can be considered.

In this work, we introduce a new method of enforcing the NMF solution of the weights matrix in the separation stage to follow certain estimated patterns. We assume we have prior statistical informations about the solution of the NMF weights matrix. The Gaussian mixture model (GMM) is used as a prior model for the valid/expected weight combination patterns that can exist in the columns of the weights matrix that are related to the nature of the source signals. Here, in the training stage, NMF is also used to decompose the spectrogram of the training data into trained basis and weights/gains matrices for each source. In this work, the trained gains matrix is used along with the trained basis matrix to represent each source. We can see the columns of the trained gains matrix as valid weight combinations that their corresponding bases in the basis matrix can jointly receive for a certain type of source signal. The columns of the trained gains matrix can be used to train a prior model that captures the statistics of the valid weight combinations that the bases can receive. The prior gain model and the trained basis matrix for each source can be used to represent each source in the separation stage. During the separation stage, the prior model can guide the NMF solution to prefer these valid weight patterns. The multivariate Gaussian mixture model (GMM) can be used to model the trained gains matrix (Grais and Erdogan,

2012b). The GMM is a rich model which captures the statistics and the correlations of the valid gain combinations for each source signal. GMMs are extensively used in speech processing applications like speech recognition and speaker verification. GMMs are used to model the multi-modal nature in speech feature vectors due to phonetic differences, speaking styles, gender, accents (Rabiner and Juang, 1993). We are conjecturing that the weight vectors of the NMF gains matrix can be considered as a feature extracted from the signal in a frame so that it can be modeled well with a GMM. The columns in the trained weights matrix are normalized, and their logarithm is then calculated and used to train the GMM prior. The basis matrix and the trained GMM prior model for the weights are jointly used as a trained representative model for the source training signals.

In the separation stage and after observing the mixed signal, NMF is used again to decompose the mixed signal spectrogram as a weighted linear combination of the trained basis vectors for the sources that are involved in the observed mixed signal. The conventional NMF solution for the weight combinations is found to minimize a predefined NMF cost function ignoring that, for each set of trained basis vectors of a certain source signal there is a set of corresponding valid weight combinations that the bases can possibly receive. In Grais and Erdogan (2012b), the prior GMM that models the valid weight combinations for each source is used to guide the NMF solution for the gains matrix during the separation stage. The priors in Grais and Erdogan (2012b) are enforced by maximizing the log-likelihood of the NMF solution with the trained prior GMMs. The priors in Grais and Erdogan (2012b) are enforced without evaluating how good the NMF solution is without using the

priors. For example, if the NMF solution without prior is not satisfactory, we would like to rely more on the priors and vice versa.

In this work, we introduce a new strategy of applying the priors on the NMF solutions of the gain matrix during the separation stage. The new strategy is based on evaluating how much the solution of the NMF gains matrix needs to rely on the prior GMMs. The NMF solutions without using priors for the weights matrix for each source during the separation stage can be seen as a deformed image, and its corresponding valid weights/gains matrix is needed to be estimated under the GMM prior. The deformation operator parameters which measure the uncertainty of the NMF solution of the weights matrix are learned directly from the observed mixed signal. The uncertainty in this work is a measurement of how far the NMF solution of the weights matrix during the separation stage is from being a valid weight pattern that is modeled in the prior GMM. The learned uncertainties are used with the minimum mean square error (MMSE) estimator to find the estimate of the valid weights matrix. The estimated valid weights matrix should also consider the minimization of the NMF cost function. To achieve these two goals, a regularized NMF is used to consider the valid weight patterns that can appear in the columns of the weights matrix while decreasing the NMF cost function. The uncertainties within MMSE estimates of the valid weight combinations are embedded in the regularized NMF cost function for this purpose. The uncertainty measurements play very important role in this work as we will show in next sections. If the uncertainty of the NMF solution of the weights matrix is high, that means the regularized NMF needs more support from the prior term. In case of low uncertainty, the regularized

NMF needs less support from the prior term. Including the uncertainty measurements in the regularization term using MMSE estimate makes the proposed regularized NMF algorithm decide automatically how much the solution should rely on the prior GMM term. This is the main advantage of the proposed regularized NMF compared to the regularization using the log-likelihood of the GMM prior or other prior distributions (Grais and Erdogan, 2012a,b; Canny, 2004). Incorporation of the uncertainties that measure the extent of distortion in the NMF weights matrix solutions in the regularization term is a main novelty of this work, which has not been seen before in the regularization literature.

The remainder of this paper is organized as follows: In section 2, we give a brief explanation about NMF. In section 3, we discuss the problem of single channel source separation and its formulation. In Section 4, we show the conventional usage of NMF in SCSS problems. Section 5 describes the needs for a regularized NMF. Sections 6 to 9 introduce the new regularized NMF and how it is used in the SCSS problem, which is the main contribution of this paper. Section 10 indicates the source signal reconstruction after NMF decomposition. In the remaining sections, we present our observations and the results of our experiments.

2. Nonnegative matrix factorization

Nonnegative matrix factorization is used to decompose any nonnegative matrix \mathbf{V} into a multiplication of a nonnegative basis matrix \mathbf{B} and a non-negative gains or weights matrix \mathbf{G} as follows:

$$\mathbf{V} \approx \mathbf{B}\mathbf{G}. \quad (1)$$

The columns of matrix \mathbf{B} contain nonnegative basis or dictionary vectors that are optimized to allow the data in \mathbf{V} to be approximated as a non-negative linear combination of its constituent vectors. Each column in the gains/weights matrix \mathbf{G} contains the set of weight combinations that the basis vectors in the basis matrix have for its corresponding column in the \mathbf{V} matrix. To solve for matrix \mathbf{B} and \mathbf{G} , different NMF cost functions can be used. For audio source separation applications, the Itakura-Saito (IS-NMF) divergence cost function (Fevotte et al., 2009) is usually used. This cost function is found to be a good measurement for the perceptual differences between different audio signals (Fevotte et al., 2009; Jauregui et al., 2011). The IS-NMF cost function is defined as:

$$\min_{\mathbf{B}, \mathbf{G}} D_{IS}(\mathbf{V} \parallel \mathbf{BG}), \quad (2)$$

where

$$D_{IS}(\mathbf{V} \parallel \mathbf{BG}) = \sum_{m,n} \left(\frac{V_{m,n}}{(\mathbf{BG})_{m,n}} - \log \frac{V_{m,n}}{(\mathbf{BG})_{m,n}} - 1 \right).$$

The IS-NMF solutions for equation (2) can be computed by alternating multiplicative updates of \mathbf{B} and \mathbf{G} (Fevotte et al., 2009; Jauregui et al., 2011) as:

$$\mathbf{B} \leftarrow \mathbf{B} \otimes \frac{\mathbf{V} \mathbf{G}^T}{(\mathbf{BG})^2}, \quad (3)$$

$$\mathbf{G} \leftarrow \mathbf{G} \otimes \frac{\mathbf{B}^T \mathbf{V}}{\mathbf{B}^T \mathbf{BG}}, \quad (4)$$

where the operation \otimes is an element-wise multiplication, all divisions and $(.)^2$ are element-wise operations. In source separation applications, IS-NMF is

used with matrices of power spectral densities of the source signals (Fevotte et al., 2009; Jaureguiberry et al., 2011).

3. Problem formulation for SCSS

In single channel source separation (SCSS) problems, the aim is to find estimates of source signals that are mixed on a single observation channel $y(t)$. This problem is usually solved in the short time Fourier transform (STFT) domain. Let $Y(t, f)$ be the STFT of $y(t)$, where t represents the frame index and f is the frequency-index. Due to the linearity of the STFT, we have:

$$Y(t, f) = S_1(t, f) + S_2(t, f), \quad (5)$$

where $S_1(t, f)$ and $S_2(t, f)$ are the unknown STFT of the first and second sources in the mixed signal. Assuming independence of the sources, we can write the power spectral density (PSD) of the measured signal as the sum of source signal PSDs as follows:

$$\sigma_y^2(t, f) = \sigma_1^2(t, f) + \sigma_2^2(t, f), \quad (6)$$

where $\sigma_y^2(t, f) = E(|Y(t, f)|^2)$. We can write the PSDs for all frames as a spectrogram matrix as follows:

$$\mathbf{Y} = \mathbf{S}_1 + \mathbf{S}_2, \quad (7)$$

where \mathbf{S}_1 and \mathbf{S}_2 are the unknown spectrograms of the source signals, and they need to be estimated using the observed mixed signal and training data for each source. The spectrogram of the measured signal \mathbf{Y} is calculated by taking the squared magnitude of the STFT of the measured signal $y(t)$.

4. Conventional NMF for SCSS

In conventional single channel source separation using NMF without regularization (Grais et al., 2012), there are two main stages to find estimates for \mathbf{S}_1 and \mathbf{S}_2 in equation (7). The first stage is the training stage and the second stage is the separation/testing stage. In the training stage, the spectrogram $\mathbf{S}^{\text{train}}$ for each source is calculated by computing the squared magnitude of the STFT of each source training signal. NMF is used to decompose the spectrogram into basis and gains matrices as follows:

$$\mathbf{S}_1^{\text{train}} \approx \mathbf{B}_1 \mathbf{G}_1^{\text{train}}, \quad \mathbf{S}_2^{\text{train}} \approx \mathbf{B}_2 \mathbf{G}_2^{\text{train}}, \quad (8)$$

the multiplicative update rules in equations (3) and (4) are used to solve for $\mathbf{B}_1, \mathbf{B}_2, \mathbf{G}_1^{\text{train}}$ and $\mathbf{G}_2^{\text{train}}$ for both sources. Within each iteration, the columns of \mathbf{B}_1 and \mathbf{B}_2 are normalized and the matrices $\mathbf{G}_1^{\text{train}}$ and $\mathbf{G}_2^{\text{train}}$ are computed accordingly. The initialization of all matrices $\mathbf{B}_1, \mathbf{B}_2, \mathbf{G}_1^{\text{train}}$ and $\mathbf{G}_2^{\text{train}}$ is done using positive random noise. After finding basis and gains matrices for each source training data, the basis matrices are used in the mixed signal decomposition as shown in the following sections. All the basis matrices \mathbf{B}_1 and \mathbf{B}_2 are kept fixed in the remaining sections in this paper.

In the separation stage after observing the mixed signal $y(t)$, NMF is used to decompose the mixed signal spectrogram \mathbf{Y} with the trained bases matrices \mathbf{B}_1 and \mathbf{B}_2 that were found from solving equation (8) as follows:

$$\mathbf{Y} \approx [\mathbf{B}_1, \mathbf{B}_2] \mathbf{G}, \quad \text{or} \quad \mathbf{Y} \approx [\mathbf{B}_1 \quad \mathbf{B}_2] \begin{bmatrix} \mathbf{G}_1 \\ \mathbf{G}_2 \end{bmatrix}, \quad (9)$$

then the corresponding spectrogram estimate for each source can be found

as:

$$\tilde{\mathbf{S}}_1 = \mathbf{B}_1 \mathbf{G}_1, \quad \tilde{\mathbf{S}}_2 = \mathbf{B}_2 \mathbf{G}_2. \quad (10)$$

Let $\mathbf{B}_{train} = [\mathbf{B}_1, \mathbf{B}_2]$. The only unknown here is the gains matrix \mathbf{G} since the matrix \mathbf{B}_{train} was found during the training stage and it is fixed in the separation stage. The matrix \mathbf{G} is a combination of two submatrices as in equation (9). NMF is used to solve for \mathbf{G} in (9) using the update rule in equation (4) and \mathbf{G} is initialized with positive random numbers.

5. Motivation for regularized NMF

The solution of the gains submatrix \mathbf{G}_1 in (9) is affected by the existence of the second source in the mixed signal. Also, \mathbf{G}_2 is affected by the first source in the mixed signal. The effect of one source into the gains matrix solution of the other source strongly depends on the energy level of each source in the mixed signal. Therefore, the estimated spectrograms $\tilde{\mathbf{S}}_1$ and $\tilde{\mathbf{S}}_2$ in equation (10) that are found from solving \mathbf{G} using the update rule in (4) may contain residual contribution from each other and other distortions. To fix this problem, more discriminative constraints must be added to the solution of each gains submatrix. The columns of the solution gains submatrix \mathbf{G}_1 and \mathbf{G}_2 should form a valid/expected weight combinations for its corresponding basis matrix of its corresponding source signal. The information about the valid weight combinations that can exist in the gains matrix for a source signal can be found in the gains matrix that was computed from the clean training data of the same source. For example, the information about valid weight combinations that can exist in the gains matrix \mathbf{G}_1 in equation (9) can be found in its training gains matrix \mathbf{G}_1^{train} in equation (8).

The columns of the trained gains matrix $\mathbf{G}_1^{\text{train}}$ represent the valid weight combinations that the basis matrix \mathbf{B}_1 can receive for the first source. Note that, the basis matrix \mathbf{B}_1 is common in the training and separation stages. The solution of the gains submatrix \mathbf{G}_1 in equation (9) should consider the prior about the valid combination that is present in its corresponding trained gains matrix $\mathbf{G}_1^{\text{train}}$ in equation (8) for the same source.

In our previous work (Grais and Erdogan, 2012b), data in the training gains matrix $\mathbf{G}_i^{\text{train}}$ for source i was modeled using a GMM. The NMF solution of the gains matrix during the separation stage was guided by the prior GMM. The GMM was learned using the logarithm of the normalized columns of the training gains matrix. The NMF solution for the gains matrix during the separation stage was enforced to increase its log-likelihood with the trained GMM prior using regularized NMF as follows:

$$C_{old} = D_{IS}(\mathbf{Y} \parallel \mathbf{B}_{train}\mathbf{G}) - R_{old}(\mathbf{G}), \quad (11)$$

where $R_{old}(\mathbf{G})$ is the weighted sum of the log-likelihoods of the log-normalized columns of the gains matrix \mathbf{G} . $R_{old}(\mathbf{G})$ was defined as follows:

$$R_{old}(\mathbf{G}) = \sum_{i=1}^2 \eta_i \Gamma_{old}(\mathbf{G}_i), \quad (12)$$

where $\Gamma_{old}(\mathbf{G}_i)$ is the log-likelihood for the submatrix \mathbf{G}_i , and η_i is the regularization parameter for source i . The regularization parameter in Grais and Erdogan (2012b) was playing two important roles. The first role was to match the scale of the IS-NMF divergence term with the scale of the log-likelihood prior term. The second role was to decide how much the regularized NMF cost function needs to rely on the prior term. The results in Grais and Erdogan (2012b) show that, when the source i has higher energy level than

the other sources, the value of its corresponding regularization parameter η_i becomes smaller than the values of other regularization parameters for the other sources. That can be reformed as follows: when the source has high energy level, the gains matrix solution of the regularized NMF in (11) rely less on the prior model and vice versa. The values of the regularization parameters in Grais and Erdogan (2012b) was chosen manually for every energy level for each source. In the cases when the conjugate prior models of the NMF solutions were used (Virtanen et al., 2008; Canny, 2004), the hyper-parameters of the prior models were also chosen manually. The conjugate prior models usually enforced on NMF solutions using a Bayesian framework (Fevotte et al., 2009; Virtanen et al., 2008; Canny, 2004). In Grais and Erdogan (2012b), it was also shown that, the hyper-parameter choices for the conjugate prior models can also depend on the energy level differences of the source signals in the mixed signal.

6. Motivation for the proposed regularized NMF

In this work, we try to use prior GMMs to guide the solution of the gains matrix during the separation stage using regularized NMF as in Grais and Erdogan (2012b) but following a totally different regularization strategy. We also try to find a way to estimate how much the solution of the regularized NMF needs to rely on the prior GMMs automatically not manually as in Grais and Erdogan (2012b). The way of finding how much the regularized NMF solution of the gains matrix needs to rely on the prior GMM is by measuring how far the statistics of the solution of the gains matrix \mathbf{G}_i in (9) is from the statistics of the solution of the valid gains matrix solution

$\mathbf{G}_i^{\text{train}}$ in (8) for source i . Recall that, the matrix $\mathbf{G}_i^{\text{train}}$ in (8) contains the weight combinations that the columns in the basis matrix \mathbf{B}_i can jointly receive for the clean data of source i . The data in $\mathbf{G}_i^{\text{train}}$ can be used as a prior information for what kinds of weight combinations that should exist in \mathbf{G}_i in (9) since the matrix \mathbf{B}_i is the same in (8) and (9). The matrix $\mathbf{G}_i^{\text{train}}$ in (8) is used to train a prior GMM for the expected (valid) weight combinations that can exist in the gains matrix for source i as in Grais and Erdogan (2012b). The solution of the gains submatrix \mathbf{G}_i in (9) can be seen as a deformed observation that needs to be restored using MMSE estimate under its corresponding GMM prior for source i . How far the statistics of the solution of the gains matrix \mathbf{G}_i is from the statistics of the solution of the valid gains matrix solution $\mathbf{G}_i^{\text{train}}$ can be seen as how much the gains submatrix \mathbf{G}_i is deformed. How much deformation exists in the gains matrix \mathbf{G}_i can be learned directly and the logarithm of this deformation is modeled using a Gaussian distribution with zero mean and a diagonal covariance matrix Ψ_i . When the deformation or the uncertainty measurement Ψ_i of the gain submatrix \mathbf{G}_i is high, we expect our target regularized NMF cost function to rely more on the prior GMM for source i and vice versa. Based on the measurement Ψ_i , the proposed NMF cost function decides automatically how much the solution of the regularized NMF needs to rely on the prior GMMs, which is a main advantage of the proposed regularized NMF over our previous work (Grais and Erdogan, 2012b). Applying the prior information on the gains matrix \mathbf{G}_i in (9) using MMSE estimate under a GMM prior using regularized NMF is the new strategy that we introduce in this paper.

In the following sections, we give more details about training the prior

GMM for the gains matrix for each source. Then, we give more details about our proposed regularized NMF using MMSE estimate to find better solution for the gains matrix in (9). In Section 8, we present our proposed regularized NMF in a general manner. In Section 8, we assume we have a trained basis matrix \mathbf{B} , a trained prior GMM for a clean gains matrix, and a gains matrix \mathbf{G} that inherited some distortion from the original matrix \mathbf{V} from solving equation (2). We introduce our proposed regularized NMF in a general fashion in Section 8 to make the idea clearer for different NMF applications like, dimensionality reduction, denosing, and other applications. The update rules that solve the proposed regularized NMF are also derived in Section 8 in a general fashion regardless of the application. The GMM in Section 8 is the trained prior GMM that captures the statistics of the valid weights combinations that should have been existed in the gains matrix \mathbf{G} . In section 9, we show how we use the proposed regularized NMF to find better solutions for the gain submatrices in equation (9) for our single channel source separation problem.

7. Training the GMM prior models

We use the gains matrices $\mathbf{G}_1^{\text{train}}$ and $\mathbf{G}_2^{\text{train}}$ in equation (8) to train prior models for the expected/valid weight patterns in the gains matrix for each source. For each matrix $\mathbf{G}_1^{\text{train}}$ and $\mathbf{G}_2^{\text{train}}$, we normalize their columns and then calculate their logarithm. The normalization in this paper is done using the Euclidean norm. The log-normalized columns are then used to train a gains prior GMM for each source. The GMM for a random variable \mathbf{x} is

defined as:

$$p(\mathbf{x}) = \sum_{k=1}^K \frac{\pi_k}{(2\pi)^{d/2} |\mathbf{\Sigma}_k|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \mathbf{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right\}, \quad (13)$$

where K is the number of Gaussian mixture components, π_k is the mixture weight, d is the vector dimension, $\boldsymbol{\mu}_k$ is the mean vector and $\mathbf{\Sigma}_k$ is the diagonal covariance matrix of the k^{th} Gaussian model. In training GMM, the expectation maximization (EM) algorithm (Dempster et al., 1977) is used to learn the GMM parameters $(\pi_k, \boldsymbol{\mu}_k, \mathbf{\Sigma}_k, \forall k = \{1, 2, \dots, K\})$ for each source given its trained gain matrix $\mathbf{G}^{\text{train}}$. The suitable value for K usually depends on the nature, dimension and the size of the available training data. We use the logarithm because it has been shown that the logarithm of a variable taking values between 0 and 1 can be modeled well by a GMM (Wessel et al., 2000). Since the main goal of the prior model is to capture the statistics of the patterns in the trained gains matrix, we use normalization to make the prior models insensitive to the energy level of the training data. The normalization makes the same prior models applicable for a wide range of energy levels and avoids the need to train a different prior model for different energy levels. By normalization we are modeling the ratio and correlation between the combination of the weights that the bases can jointly receive.

8. The proposed regularized NMF

The goal of regularized NMF is to incorporate prior information on the solution matrices \mathbf{B} and \mathbf{G} . In this work, we enforce a statistical prior information on the solution of the gains/weights matrix \mathbf{G} only. We need the solution of the gains matrix \mathbf{G} to minimize the IS-divergence cost function

in equation (2), and the columns of the gains matrix \mathbf{G} should form valid weight combinations under a prior GMM model.

The most used strategy for incorporating a prior is by maximizing the likelihood of the solution under the prior model while minimizing the NMF divergence at the same time. To achieve this, we usually add these two objectives in a single cost function. In Grais and Erdogan (2012b), a GMM was used as the prior model for the gains matrix, and the solution of the gains matrix was encouraged to increase its log-likelihood with the prior model using this regularized NMF cost function. The regularization parameters in Grais and Erdogan (2012b) were the only tools to control how much the regularized NMF relies on the prior models based on the energy differences of the sources in the mixed signal. The values of the regularization parameters were changed manually in that work.

Gaussian mixture model is a very general prior model where we can see the means of the GMM mixture components as “valid templates” that were observed in the training data. Even, Parzen density priors (Kim et al., 2007) can be seen under the same framework. In Parzen density prior estimation, training examples are seen as “valid templates” and a fixed variance is assigned to each example. In GMM priors, we learn the templates as cluster means from training data and we can also estimate the cluster variances from the data. We can think of the GMM prior as a way to encourage the use of valid templates or cluster means in the NMF solution during the separation stage. This view of the GMM prior will be helpful in understanding the MMSE estimate method we introduce in this paper.

We can find a way of measuring how far the conventional NMF (NMF

without prior) solution is from the trained templates in the prior GMM and call this the error term. Based on this error, the regularized NMF can decide automatically how much the solution of the NMF needs help from the prior model. If the conventional NMF solution is far from the templates then the regularized NMF will rely more on the prior model. If the conventional NMF solution is close to the templates then the regularized NMF will rely less on the prior model. By deciding automatically how much the regularized NMF needs to rely on the prior we conjecture that, we do not need to manually change the values for the regularization parameter based on the energy differences of the sources in the mixed signal ¹ to improve the performance of NMF as in Grais and Erdogan (2012b).

We use the following way of measuring how far the conventional NMF solution is from the prior templates: We can see the solution of the conventional NMF as distorted observations of a true/valid template. Given the prior GMM templates, we can learn a probability distribution model for the distortion that captures how far the observations in the conventional gains matrix is from the prior GMM. The distortion or the error model can be seen as a summary of the distortion that exists in all columns in the gains matrix of the NMF solution.

Based on the prior GMM and the trained distortion model, we can find a better estimate for the desired observation for each column in the distorted gains matrix. We can mathematically formulate this by seeing the solution matrix \mathbf{G} that only minimizes the cost function in equation (2) as a distorted

¹In this paper, the regularization parameters are chosen once and kept fixed regardless of the energy differences of the source signals

image where its restored image needs to be estimated. The columns of the matrix \mathbf{G} are normalized using the ℓ^2 norm and their logarithm is then calculated. Let the log-normalized column n namely $(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2})$ of the gains matrix be \mathbf{q}_n . The vector \mathbf{q}_n is treated as a distorted observation as:

$$\mathbf{q}_n = \mathbf{x}_n + \mathbf{e}, \quad (14)$$

where \mathbf{x}_n is the logarithm of the unknown desired pattern that corresponds to the observation \mathbf{q}_n and needs to be estimated under a prior GMM, \mathbf{e} is the logarithm of the deformation operator, which is modeled by a Gaussian distribution with zero mean and diagonal covariance matrix Ψ as $\mathcal{N}(\mathbf{e}|\mathbf{0}, \Psi)$. The GMM prior model for the gains matrix is trained using log-normalized columns of the trained gains matrix from training data as shown for example in Section 7. The uncertainty Ψ is trained directly from all the log-normalized columns of the gains matrix $\mathbf{q} = \{\mathbf{q}_1, \dots, \mathbf{q}_n, \dots, \mathbf{q}_N\}$, where N is the number of columns in the matrix \mathbf{G} . The uncertainty Ψ can be iteratively learned using the expectation maximization (EM) algorithm. Given the prior GMM parameters which are considered fixed here, the update of Ψ is found based on the sufficient statistics $\hat{\mathbf{z}}_n$ and \hat{R}_n as follows (Rosti and Gales, 2001, 2004; Ghahramani and Hinton, 1997) [Appendix A]:

$$\Psi = \text{diag} \left\{ \frac{1}{N} \sum_{n=1}^N \left(\mathbf{q}_n \mathbf{q}_n^T - \mathbf{q}_n \hat{\mathbf{z}}_n^T - \hat{\mathbf{z}}_n \mathbf{q}_n^T + \hat{R}_n \right) \right\}, \quad (15)$$

where the “diag” operator sets all the off-diagonal elements of a matrix to zero, N is the number of columns in matrix \mathbf{G} , and the sufficient statistics $\hat{\mathbf{z}}_n$ and \hat{R}_n can be updated using Ψ from the previous iteration as follows:

$$\hat{\mathbf{z}}_n = \sum_{k=1}^K \gamma_{kn} \hat{\mathbf{z}}_{kn}, \quad (16)$$

and

$$\hat{\mathbf{R}}_n = \sum_{k=1}^K \gamma_{kn} \hat{\mathbf{R}}_{kn}, \quad (17)$$

where

$$\gamma_{kn} = \left[\frac{\pi_k \mathcal{N}(\mathbf{q}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k + \boldsymbol{\Psi})}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{q}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j + \boldsymbol{\Psi})} \right], \quad (18)$$

$$\hat{\mathbf{R}}_{kn} = \boldsymbol{\Sigma}_k - \boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1} \boldsymbol{\Sigma}_k^T + \hat{\mathbf{z}}_{kn} \hat{\mathbf{z}}_{kn}^T, \quad (19)$$

and

$$\hat{\mathbf{z}}_{kn} = \boldsymbol{\mu}_k + \boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1} (\mathbf{q}_n - \boldsymbol{\mu}_k). \quad (20)$$

$\boldsymbol{\Psi}$ is considered as a general uncertainty measurement over all the observations in matrix \mathbf{G} . $\boldsymbol{\Psi}$ can be seen as a model that summarizes the deformation that exists in all columns in the gains matrix \mathbf{G} .

Given the GMM prior parameters and the uncertainty measurement $\boldsymbol{\Psi}$, the MMSE estimate of each pattern \mathbf{x}_n given its observation \mathbf{q}_n under the observation model in equation (14) can be found similar to Rosti and Gales (2001, 2004), and Ghahramani and Hinton (1997) as in Appendix A as follows:

$$f(\mathbf{q}_n) = \sum_{k=1}^K \gamma_{kn} [\boldsymbol{\mu}_k + \boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1} (\mathbf{q}_n - \boldsymbol{\mu}_k)] = \hat{\mathbf{x}}_n, \quad (21)$$

where

$$\gamma_{kn} = \left[\frac{\pi_k \mathcal{N}(\mathbf{q}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k + \boldsymbol{\Psi})}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{q}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j + \boldsymbol{\Psi})} \right]. \quad (22)$$

The value of $\boldsymbol{\Psi}$ in the term $\boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1}$ in equation (21) plays an important role in this framework. When the entries of the uncertainty $\boldsymbol{\Psi}$ are very small comparing to their corresponding entries in $\boldsymbol{\Sigma}_k$ for a certain active GMM component k , the term $\boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1}$ tends to be the identity matrix, and MMSE estimate in (21) will be the observation \mathbf{q}_n . When the entries of

the uncertainty Ψ are very high comparing to their corresponding entries in Σ_k for a certain active GMM component k , the term $\Sigma_k (\Sigma_k + \Psi)^{-1}$ tends to be a zeros matrix, and MMSE estimate will be the weighted sum of prior templates $\sum_{k=1}^K \gamma_{kn} \mu_k$. In most cases γ_{kn} tends to be close to one for one Gaussian component, and close to zero for the other components in a large dimension space. This makes the MMSE estimate in the case of high Ψ to be one of the mean vectors in the prior GMM, which is considered as a template pattern for the valid observation. We can rephrase this as follows: When the uncertainty of the observations \mathbf{q} is high, the MMSE estimate of \mathbf{x} , relies more on the prior GMM of \mathbf{x} . When the uncertainty of the observations \mathbf{q} is low, the MMSE estimate of \mathbf{x} , relies more on the observation \mathbf{q}_n . In general, the MMSE solution of \mathbf{x} lies between the observation \mathbf{q}_n and one of the templates in the prior GMM. The term $\Sigma_k (\Sigma_k + \Psi)^{-1}$ controls the distance between $\hat{\mathbf{x}}_n$ and \mathbf{q}_n and also the distance between $\hat{\mathbf{x}}_n$ and one of the template μ_k assuming that $\gamma_{kn} \approx 1$ for a Gaussian component k .

The model in equation (14) expresses the normalized columns of the gains matrix as a distorted image with a multiplicative deformation diagonal matrix. For the normalized gain columns $\frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}$ of \mathbf{G} there is a deformation matrix \mathbf{E} with log-normal distribution that is applied to the correct pattern that we need to estimate $\hat{\mathbf{g}}_n$ as follows:

$$\frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} = \mathbf{E} \hat{\mathbf{g}}_n. \quad (23)$$

The uncertainty for \mathbf{E} is represented in its covariance matrix Ψ . For the distorted matrix \mathbf{G} we find its corresponding MMSE estimate for its log-normalized columns $\hat{\mathbf{G}}$. Another reason for working in logarithm domain is that, the gains are constrained to be nonnegative and the MMSE estimate

can be negative so the logarithm of the normalized gains is an unconstrained variable that we can work with. The estimated weight patterns in $\hat{\mathbf{G}}$ that are corresponding to the MMSE estimates for the correct patterns do not consider minimizing the NMF cost function in equation (2), which is still the main goal. We need the solution of \mathbf{G} to consider the pattern shape priors on the solution of the gains matrix, and also considers the reconstruction error of the NMF cost function. To consider the combination of the two objectives, we consider using the regularized NMF. We add a penalty term to the NMF-divergence cost function. The penalty term tries to minimize the distance between the solution of log-normalized columns of \mathbf{g}_n with its corresponding MMSE estimate $f(\mathbf{g}_n)$ as follows:

$$\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} \approx f\left(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}\right) \quad \text{or} \quad \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} \approx \exp\left(f\left(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}\right)\right). \quad (24)$$

The regularized IS-NMF cost function is defined as follows:

$$C = D_{IS}(\mathbf{V} \parallel \mathbf{B}\mathbf{G}) + \alpha L(\mathbf{G}), \quad (25)$$

where

$$L(\mathbf{G}) = \sum_{n=1}^N \left\| \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} - \exp\left(f\left(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}\right)\right) \right\|_2^2, \quad (26)$$

$f\left(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}\right)$ is the MMSE estimate defined in equation (21), and α is a regularization parameter. The regularized NMF can be rewritten in more details as

$$C = \sum_{m,n} \left(\frac{\mathbf{V}_{m,n}}{(\mathbf{B}\mathbf{G})_{m,n}} - \log \frac{\mathbf{V}_{m,n}}{(\mathbf{B}\mathbf{G})_{m,n}} - 1 \right) + \alpha \sum_{n=1}^N \left\| \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} - \exp\left(\sum_{k=1}^K \gamma_{kn} \left[\boldsymbol{\mu}_k + \boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1} \left(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} - \boldsymbol{\mu}_k \right) \right] \right) \right\|_2^2. \quad (27)$$

In equation (27), the MMSE estimate of the desired patterns of the gains matrix is embedded in the regularized NMF cost function. The first term

in (27), decreases the reconstruction error between \mathbf{V} and $\mathbf{B}\mathbf{G}$. Given Ψ , we can forget for a while the MMSE estimate concept that led us to our target regularized NMF cost function in (27) and see equation (27) as an optimization problem. We can see from (27) that, if the distortion measurement parameter Ψ is high, the regularized nonnegative matrix factorization solution for the gains matrix will rely more on the prior GMM for the gains matrix. If the distortion parameter Ψ is low, the regularized nonnegative matrix factorization solution for the gains matrix will be close to the ordinary NMF solution for the gains matrix without considering any prior. The second term in equation (27) is ignored in the case of zero uncertainty Ψ . In case of high values of Ψ , the second term encourages to decrease the distance between each normalized column $\frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}$ in \mathbf{G} with a corresponding prior template $\exp(\boldsymbol{\mu}_k)$ assuming that $\gamma_{kn} \approx 1$ for a certain Gaussian component k . For different values Ψ , the penalty term decreases the distance between each $\frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}$ and an estimated pattern that lies between a prior template and $\frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}$. The term $(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} - \boldsymbol{\mu}_k)$ in (27) measures how far each log-normalized column in the gains matrix is from a valid template $\boldsymbol{\mu}_k$. Under the assumption $\gamma_{kn} \approx 1$ for a certain Gaussian component k , the second term in (27) is also ignored when the observation $\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}$ form a valid pattern ($\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} = \boldsymbol{\mu}_k$). How far each log-normalized column in the gains matrix is from a valid template decides how much influence the MMSE estimate prior term has to the solution of (27) for each observation.

The multiplicative update rule for \mathbf{B} in (27) is still the same as in equation (3). The multiplicative update rule for \mathbf{G} can be found by following the same procedures as in Virtanen (2007); Bertin et al. (2010); Grais and Erdogan

(2012b). The gradient with respect to \mathbf{G} of the cost function $\nabla_G C$ can be expressed as a difference of two positive terms $\nabla_G^+ C$ and $\nabla_G^- C$ as follows:

$$\nabla_G C = \nabla_G^+ C - \nabla_G^- C. \quad (28)$$

The cost function is shown to be nonincreasing under the update rule (Virtanen, 2007; Bertin et al., 2010):

$$\mathbf{G} \leftarrow \mathbf{G} \otimes \frac{\nabla_G^- C}{\nabla_G^+ C}, \quad (29)$$

where the operations \otimes and division are element-wise as in equation (4). We can write the gradients as:

$$\nabla_G C = \nabla_G D_{IS} + \alpha \nabla_G L(\mathbf{G}), \quad (30)$$

where $\nabla_G L(\mathbf{G})$ is a matrix with the same size of \mathbf{G} . The gradient for the IS-NMF and the gradient of the prior term can also be expressed as a difference of two positive terms as follows:

$$\nabla_G D_{IS} = \nabla_G^+ D_{IS} - \nabla_G^- D_{IS}, \quad (31)$$

and

$$\nabla_G L(\mathbf{G}) = \nabla_G^+ L(\mathbf{G}) - \nabla_G^- L(\mathbf{G}). \quad (32)$$

We can rewrite equations (28, 30) as:

$$\nabla_G C = (\nabla_G^+ D_{IS} + \alpha \nabla_G^+ L(\mathbf{G})) - (\nabla_G^- D_{IS} + \alpha \nabla_G^- L(\mathbf{G})). \quad (33)$$

The final update rule in equation (29) can be written as follows:

$$\mathbf{G} \leftarrow \mathbf{G} \otimes \frac{\nabla_G^- D_{IS} + \alpha \nabla_G^- L(\mathbf{G})}{\nabla_G^+ D_{IS} + \alpha \nabla_G^+ L(\mathbf{G})}, \quad (34)$$

where

$$\nabla_G D_{IS} = \mathbf{B}^T \frac{\mathbf{1}}{\mathbf{B}\mathbf{G}} - \mathbf{B}^T \frac{\mathbf{V}}{(\mathbf{B}\mathbf{G})^2}, \quad (35)$$

$$\nabla_G^- D_{IS} = \mathbf{B}^T \frac{\mathbf{V}}{(\mathbf{B}\mathbf{G})^2}, \quad \text{and} \quad \nabla_G^+ D_{IS} = \mathbf{B}^T \frac{\mathbf{1}}{\mathbf{B}\mathbf{G}}. \quad (36)$$

Note that, in calculating the gradients $\nabla_G^+ L(\mathbf{G})$ and $\nabla_G^- L(\mathbf{G})$, the term γ_{kn} is also a function of \mathbf{G} . The gradients $\nabla_G^+ L(\mathbf{G})$ and $\nabla_G^- L(\mathbf{G})$ are calculated in Appendix B. Since all the terms in equation (34) are nonnegative, then the values of \mathbf{G} of the update rule (34) are nonnegative.

9. The proposed regularized NMF for SCSS

In this section, we are back to the single channel source separation problem to find a better solution to equation (9). Figure 1 shows the flow chart that summarizes all stages of applying our proposed regularized NMF method for SCSS problems. Given the trained basis matrices $\mathbf{B}_1, \mathbf{B}_2$ that were computed from solving (8), and the trained gain prior GMM for each source from Section 7, we try to apply the proposed regularized NMF cost function in Section 8 to find better solution for the gain submatrices in equation (9). The bases matrix $\mathbf{B}_{train} = [\mathbf{B}_1, \mathbf{B}_2]$ is still fixed here, we just need to update the gains matrix \mathbf{G} in (9). The normalized columns of the submatrices \mathbf{G}_1 and \mathbf{G}_2 in equation (9) can be seen as deformed images as in equation (23) and their restored images are needed to be estimated. First, we need to learn the uncertainties parameters Ψ_1 and Ψ_2 for the deformation operators \mathbf{E}_1 and \mathbf{E}_2 respectively for each image as shown in learning the uncertainties stage in Figure 1. The columns of the submatrix \mathbf{G}_1 are normalized and

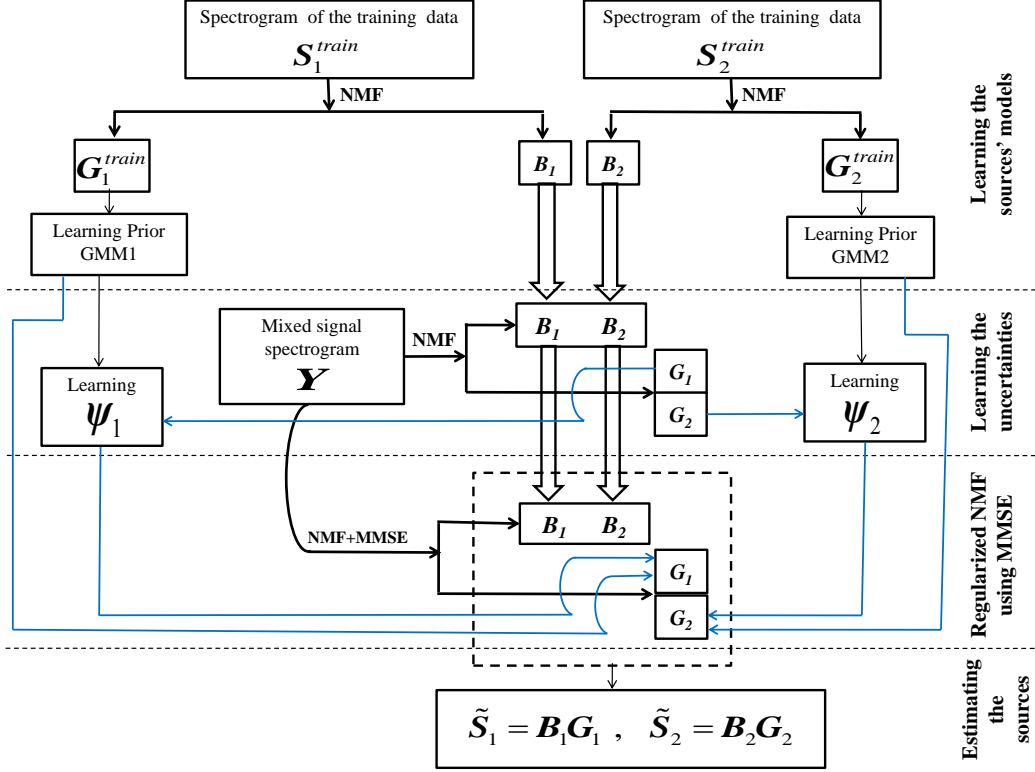


Figure 1: The flow chart of using regularized NMF with MMSE estimates under GMM priors for SCSS. The term NMF+MMSE means regularized NMF using MMSE estimates under GMM priors.

their logarithm are calculated and used with the trained GMM prior parameters for the first source to estimate Ψ_1 iteratively using the EM algorithm in equations (15) to (20). The log-normalized columns “ $\log \frac{g_n}{\|g_n\|_2}$ ” of G_1 can be seen as q_n in equations (15) to (20). We repeat the same procedures to calculate Ψ_2 using the log-normalized columns of G_2 and the prior GMM for the second source. The uncertainties Ψ_1 and Ψ_2 can also be seen as measurements of the remaining distortion from one source into another source, which also depends on the mixing ratio between the two sources. For example, if

the first source has higher energy than the second source in the mixed signal, we expect the values of Ψ_2 to be higher than the values in Ψ_1 and vice versa. After calculating the uncertainty parameters for both sources Ψ_1 and Ψ_2 , we use the regularized NMF in (25) to solve for \mathbf{G} with the prior GMMs for both sources and the estimated uncertainties Ψ_1 and Ψ_2 as follows:

$$C = D_{IS}(\mathbf{Y} \parallel \mathbf{B}_{train}\mathbf{G}) + R(\mathbf{G}), \quad (37)$$

where

$$R(\mathbf{G}) = \alpha_1 L_1(\mathbf{G}_1) + \alpha_2 L_2(\mathbf{G}_2), \quad (38)$$

$L_1(\mathbf{G}_1)$ is defined as in equation (26) for the first source, $L_2(\mathbf{G}_2)$ is for the second source, α_1 , and α_2 are their corresponding regularization parameters. The update rule in equation (34) can be used to solve for \mathbf{G} after modifying it as follows:

$$\mathbf{G} \leftarrow \mathbf{G} \otimes \frac{\nabla_G^- D_{IS} + \nabla_G^- R(\mathbf{G})}{\nabla_G^+ D_{IS} + \nabla_G^+ R(\mathbf{G})}, \quad (39)$$

where $\nabla_G^+ R(\mathbf{G})$ and $\nabla_G^- R(\mathbf{G})$ are nonnegative matrices with the same size of \mathbf{G} and they are combinations of two submatrices as follows:

$$\nabla_G^- R(\mathbf{G}) = \begin{bmatrix} \alpha_1 \nabla_G^- L(\mathbf{G}_1) \\ \alpha_2 \nabla_G^- L(\mathbf{G}_2) \end{bmatrix}, \quad \nabla_G^+ R(\mathbf{G}) = \begin{bmatrix} \alpha_1 \nabla_G^+ L(\mathbf{G}_1) \\ \alpha_2 \nabla_G^+ L(\mathbf{G}_2) \end{bmatrix}, \quad (40)$$

where $\nabla_G^+ L(\mathbf{G}_1)$, $\nabla_G^- L(\mathbf{G}_1)$, $\nabla_G^+ L(\mathbf{G}_2)$, and $\nabla_G^- L(\mathbf{G}_2)$ are calculated as in section 8 for each source.

The normalization of the columns of the gain matrices are used in the prior term $R(\mathbf{G})$ and its gradient terms only. The general solution for the gains matrix of equation (37) at each iteration is not normalized. The normalization is done only in the prior term since the prior models have been

trained by normalized data before. Normalization is also useful in cases where the source signals occur with different energy levels from each other in the mixed signal. Normalizing the training and testing gain matrices gives the prior models a chance to work with any energy level that the source signals can take in the mixed signal regardless of the energy levels of the training signals.

The regularization parameters in (38) have only one role. They are chosen to match the scale between the NMF divergence term and the MMSE estimate prior term in the regularized NMF cost function in (37). There is no need to change the values of the regularization parameters according to the energy differences of the source signals in the mixed signal as in Grais and Erdogan (2012b). Reasonable values for the regularization parameters are chosen manually and kept fixed in this work. Another main difference between the regularized NMF in Grais and Erdogan (2012b) that is shown in equation (11) and the proposed regularized NMF in this paper is related to the training procedures for the source models. In both works, the main aim of the training stage is to train the basis matrices and the gains prior GMMs for the source signals. In Grais and Erdogan (2012b), to match between the way the trained models were used during training with the way they were used during separation, the basis matrices and the prior GMM parameters were learned jointly using the regularized NMF cost function in (11). The joint training for the source models was introduced in Grais and Erdogan (2012b) to improve the separation performance. In joint training, after updating the gains matrix at each NMF iteration using the gain update rule for the regularized NMF in (11), the GMM parameters were then updated (re-

trained). Since, we needed to update (retrain) the GMM parameters at each NMF iteration, joint training slowed down the training of the source models in Grais and Erdogan (2012b). Another problem of using joint training is that, we had other regularization parameters during the training stage that needed to be chosen. Using joint training duplicates the number of the regularization parameters that need to be chosen. Choosing the regularization parameters in Grais and Erdogan (2012b) was done using validation data. That means, in Grais and Erdogan (2012b) we had to train many source models (basis matrix and prior GMM) for different regularization parameter values. Then, we chose the best combination for the regularization parameter values in training and separation stages that gave the best results during the separation stage. In the case of using MMSE estimate regularization for NMF, we do not need to use joint training. In this paper, we do not need to consider solving the regularized NMF in (27) during the training stage to solve (8). In the training stage, the training data for each source is assumed to be clean data. Since the spectrogram of each source training data represents clean source data, the NMF solution for the gains matrix can not be seen as a distorted image. Therefore, the deformation measurement parameter Ψ^{train} is a matrix of zeros. When $\Psi^{train} = \mathbf{0}$, the MMSE estimates prior term in (27) will disappear because $\sum_{k=1}^K \gamma_{kn} = 1$. Then, the regularized NMF (27) becomes just NMF. That means, we do not need to use the regularized NMF during the training stage which is not the case in Grais and Erdogan (2012b). Here in the training stage, we just need to use IS-NMF to decompose the spectrogram of the training data into trained basis and gains matrices. After the trained gains matrix is computed, it is used to train the

prior GMM as shown in Sections 4 and 7.

10. Source signals reconstruction

After finding the suitable solution for the gains matrix \mathbf{G} in Section 9, the initial estimated spectrograms $\tilde{\mathbf{S}}_1$ and $\tilde{\mathbf{S}}_2$ can be calculated from (10) and then used to build spectral masks as follows:

$$\mathbf{H}_1 = \frac{\tilde{\mathbf{S}}_1}{\tilde{\mathbf{S}}_1 + \tilde{\mathbf{S}}_2}, \quad \mathbf{H}_2 = \frac{\tilde{\mathbf{S}}_2}{\tilde{\mathbf{S}}_1 + \tilde{\mathbf{S}}_2}, \quad (41)$$

where the divisions are done element-wise. The final estimate of each source STFT can be obtained as follows:

$$\hat{S}_1(t, f) = \mathbf{H}_1(t, f) Y(t, f), \quad \hat{S}_2(t, f) = \mathbf{H}_2(t, f) Y(t, f), \quad (42)$$

where $Y(t, f)$ is the STFT of the observed mixed signal in equation (5), $\mathbf{H}_1(t, f)$ and $\mathbf{H}_2(t, f)$ are the entries at row f and column t of the spectral masks \mathbf{H}_1 and \mathbf{H}_2 respectively. The spectral mask entries scale the observed mixed signal STFT entries according to the contribution of each source in the mixed signal. The spectral masks can be seen as the Wiener filter as in Fevotte et al. (2009). The estimated source signals $\hat{s}_1(t)$ and $\hat{s}_2(t)$ can be found by using inverse STFT of their corresponding STFTs $\hat{S}_1(t, f)$ and $\hat{S}_2(t, f)$.

11. Experiments and Discussion

We applied the proposed algorithm to separate a speech signal from a background piano music signal. Our main goal was to get a clean speech

signal from a mixture of speech and piano signals. We simulated our algorithm on a collection of speech and piano data at 16kHz sampling rate. For speech data, we used the training and testing male speech data from the TIMIT database. For music data, we downloaded piano music data from the piano society web site (URL, 2009a). We used 12 pieces with approximate 50 minutes total duration from different composers but from a single artist for training and left out one piece for testing. The PSD for the speech and music data were calculated by using the STFT: A Hamming window with 480 points length and 60% overlap was used and the FFT was taken at 512 points, the first 257 FFT points only were used since the conjugate of the remaining 255 points are involved in the first points. We trained 128 basis vectors for each source, which makes the size of $\mathbf{B}_{\text{speech}}$ and $\mathbf{B}_{\text{music}}$ matrices to be 257×128 , hence, the vector dimension $d = 128$ in equation (13) for both sources. The mixed data was formed by adding random portions of the test music file to 20 speech files from the test data of the TIMIT database at different speech-to-music ratio (SMR) values in dB. The audio power levels of each file were found using the “audio voltmeter” program from the G.191 ITU-T STL software suite (URL, 2009b). For each SMR value, we obtained 20 mixed utterances this way. We used the first 10 utterances as a validation set to choose reasonable values for the regularization parameters α_{speech} and α_{music} and the number of Gaussian mixture components K . The other 10 mixed utterances were used for testing.

Performance measurement of the separation algorithm was done using the signal to noise ratio (SNR). The average SNR over the 10 test utterances for each SMR case are reported. We also used signal to interference ratio

(SIR), which is defined as the ratio of the target energy to the interference error due to the music signal only (Vincent et al., 2006).

Table 1 shows SNR and SIR of the separated speech signal using NMF with different values of the number of Gaussian mixture components K and fixed regularization parameters $\alpha_{\text{speech}} = \alpha_{\text{music}} = 1$. The first column of the Table, shows the separation results of using just NMF without any prior.

Table 1: SNR and SIR in dB for the estimated speech signal with regularization parameters $\alpha_{\text{speech}} = \alpha_{\text{music}} = 1$ and different number of Gaussian mixture components K .

SMR dB	No prior		$K = 1$		$K = 4$		$K = 8$		$K = 16$		$K = 32$	
	SNR	SIR	SNR	SIR	SNR	SIR	SNR	SIR	SNR	SIR	SNR	SIR
-5	2.88	4.86	3.31	5.71	3.61	6.58	4.24	8.07	4.76	10.07	4.27	8.39
0	5.50	8.70	5.74	9.31	5.90	9.99	6.32	11.61	6.45	13.02	6.54	12.42
5	8.37	12.20	8.46	12.40	8.55	12.98	8.74	14.13	8.73	15.62	8.69	14.51

As we can see from the Table, the proposed regularized NMF algorithm improves the separation performance for challenging SMR cases compared with using just NMF without priors. Increasing the number of Gaussian mixture components K improves the separation performance until $K = 16$. From the shown results, $K = 16$ seems to be a good choice for the given data sets. The best choice for K usually depends on the nature and the size of the training data. For example, for speech signal in general there are variety of phonetic differences, gender, speaking styles, accents, which raises the necessity for using many Gaussian components.

Comparison with other priors

In this section we compared our proposed method of using MMSE estimates under GMM prior on the solution of NMF with two other prior methods. The first prior is the sparsity prior and the second prior is enforced by maximizing the loglikelihood under GMM prior distribution.

In the sparsity prior, the NMF solution of the gains matrix was enforced to be sparse (Virtanen and Cemgil, 2009; Schmidt and Olsson, 2006). The sparse NMF is defined as

$$C(G) = D_{IS}(\mathbf{Y} \parallel \mathbf{BG}) + \lambda \sum_{m,n} \mathbf{G}_{m,n}, \quad (43)$$

where λ is the regularization parameter. The gain update rule of \mathbf{G} can be found as follows:

$$\mathbf{G} \leftarrow \mathbf{G} \otimes \frac{\mathbf{B}^T \frac{\mathbf{Y}}{(\mathbf{BG})^2}}{\mathbf{B}^T \frac{1}{\mathbf{BG}} + \lambda}. \quad (44)$$

Enforcing sparsity on the NMF solution of the gains matrix is equivalent to model the prior of the gains matrix using exponential distribution with parameter λ (Virtanen and Cemgil, 2009). The update rule in equation (44) is found based on maximizing the likelihood of the gains matrix under the exponential prior distribution.

The second method of enforcing prior on the NMF solution is by using GMM gain prior (Grais and Erdogan, 2012a,b). The NMF solution for the gains matrix is enforced to increase its log-likelihood with the trained GMM prior as follows:

$$C = D_{IS}(\mathbf{Y} \parallel \mathbf{BG}) - R_2(\mathbf{G}), \quad (45)$$

where $R_2(\mathbf{G})$ is the weighted sum of the log-likelihoods of the log-normalized columns of the gains matrix \mathbf{G} . $R_2(\mathbf{G})$ can be written as follows:

$$R_2(\mathbf{G}) = \sum_{i=1}^2 \eta_i \Gamma(\mathbf{G}_i), \quad (46)$$

where $\Gamma(\mathbf{G}_i)$ is the log-likelihood for the submatrix \mathbf{G}_i for source i .

In sparsity and GMM based log-likelihood prior methods, to match between the used update rule for the gains matrix during training and separation, the priors were enforced during both training and separation stages. In sparse NMF we used sparsity constraints during training and separation stages. In regularized NMF with GMM based log-likelihood prior we trained the NMF bases and the prior GMM parameters jointly as shown in Grais and Erdogan (2012b).

In the sparse NMF case, we got best results when $\lambda = 0.0001$ for both sources in the training and separation stages. In the case of enforcing the gains matrix to increase the log-likelihood under GMM prior (Grais and Erdogan, 2012b) we got the best results when $\eta = 1$ in the training and $\eta = 0.1$ in the separation stage. The number of Gaussian components was $K = 4$ for both sources. It is important to note that, in the case of using MMSE under GMM prior there is no need to enforce prior during training since the uncertainty measurements during training are assumed to be zeros since the training data are clean signals. When the uncertainty is zero, then the regularized NMF in case of MMSE under GMM prior is the same as the NMF cost function, then the update rule for the gains matrix in the training stage is the same as the update rule in the case of using just NMF.

Figures 2 and 3 show the SNR and SIR for the different type of prior

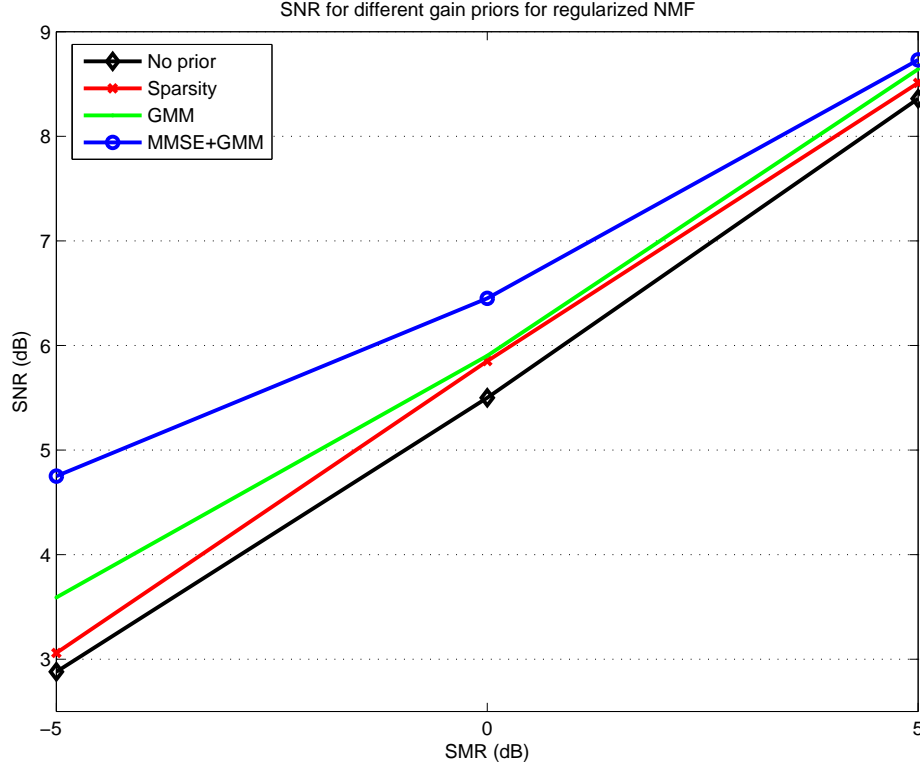


Figure 2: The effect of using different prior models on the gains matrix. The black line for using no prior case, the red line for using the exponential distribution prior, the green line is for maximizing the gains matrix likelihood with the GMM prior, and the blue line is for using MMSE under GMM as a prior

models. The black line shows the separation performance in the case of no prior is used. The red line shows the performance for the case of using sparse NMF. The green line shows the performance in the case of enforcing the gains matrix to increase its likelihood with the prior GMM. The blue line shows the separation performance in the case of using MMSE estimate under GMM prior that is proposed in this paper. As we can see, the proposed method of enforcing prior on the gains matrix using MMSE estimate under GMM

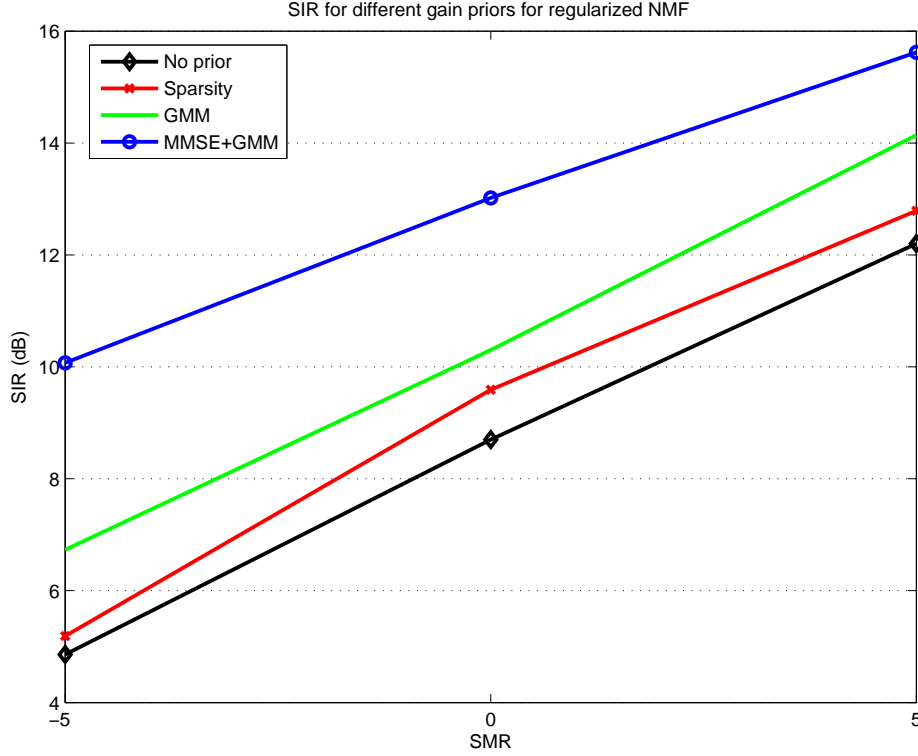


Figure 3: The effect of using different prior models on the gains matrix with the same color map of the previous figure.

prior gives the best performance comparing with the other methods. The used MMSE estimates prior in this work gives better results than the GMM likelihood method (Grais and Erdogan, 2012b) because of the measurements of the uncertainties in the MMSE under GMM case. The uncertainties work as feedback measurements that adjust the needs to the prior based on the amount of distortion in the gains matrix during the separation stage.

Comparing the relative improvements in dB that we got in this paper with the achieved improvements in other works (Wilson et al., 2008b,a; Virtanen and Cemgil, 2009; Virtanen, 2007) we can see that the, improvements in this

paper can be considered to be high.

12. CONCLUSION

In this work, we introduced a new regularized NMF algorithm. The NMF solution for the gains matrix was guided by the MMSE estimate under a GMM prior where the uncertainty of the observed mixed signal was learned online from the observed data. The proposed algorithm can be extended for better measurements of the distortion in the observed signal by embedding more parameters in equation (14) that can be learned online from the observed signal.

13. Acknowledgements

This research is partially supported by Turk Telekom Group Research and Development, project entitled “Single Channel Source Separation”, grant number 3014-06, support year 2012.

References

- Bertin, N., Badeau, R., Vincent, E., 2009. Fast Bayesian NMF algorithms enforcing harmonicity and temporal continuity in polyphonic music transcription, in: IEEE workshop on applications of signal processing to audio and acoustics.
- Bertin, N., Badeau, R., Vincent, E., 2010. Enforcing harmonicity and smoothness in bayesian nonnegative matrix factorization applied to polyphonic music transcription. IEEE Transactions, Audio, speech, and language processing 18, 538–549.

- Canny, J., 2004. GaP: a factor model for discrete data, in: Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.
- Cemgil, A.T., Dikmen, O., 2007. Conjugate Gamma Markov random fields for modelling nonstationary sources, in: International Conference on Independent Component Analysis and Signal Separation.
- Chen, Z., Cichocki, A., Rutkowski, T.M., 2006. Constrained non-negative matrix factorization method for EEG analysis in early detection of alzheimers disease, in: IEEE International Conference on Acoustics, Speech, and Signal Processing.
- Cichocki, A., Zdunek, R., Amari, S., 2006. New algorithms for nonnegative matrix factorization in applications to blind source separation, in: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP).
- Dempster, A., Laird, N., Rubin, D., 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society* .
- Fevotte, C., Bertin, N., Durrieu, J.L., 2009. Nonnegative matrix factorization with the itakura-saito divergence. With application to music analysis. *Neural Computation* 21, 793–830.
- Ghahramani, Z., Hinton, G.E., 1997. The EM algorithm for mixtures of factor analyzers. Technical Report. CRG-TR-96-1, University of Toronto, Canada.

- Grais, E.M., Erdogan, H., 2011a. Adaptation of speaker-specific bases in non-negative matrix factorization for single channel speech-music separation, in: Annual Conference of the International Speech Communication Association (INTERSPEECH).
- Grais, E.M., Erdogan, H., 2011b. Single channel speech music separation using nonnegative matrix factorization and spectral masks, in: International Conference on Digital Signal Processing.
- Grais, E.M., Erdogan, H., 2011c. Single channel speech music separation using nonnegative matrix factorization with sliding window and spectral masks, in: Annual Conference of the International Speech Communication Association (INTERSPEECH).
- Grais, E.M., Erdogan, H., 2012a. Gaussian mixture gain priors for regularized nonnegative matrix factorization in single-channel source separation, in: Annual Conference of the International Speech Communication Association (INTERSPEECH).
- Grais, E.M., Erdogan, H., 2012b. Regularized nonnegative matrix factorization using gaussian mixture priors for supervised single channel source separation. Computer Speech and Language <http://dx.doi.org/10.1016/j.csl.2012.09.002> .
- Grais, E.M., Erdogan, H., 2012c. Spectro-temporal post-smoothing in NMF based single-channel source separation, in: European Signal Processing Conference (EUSIPCO).

- Grais, E.M., Topkaya, I.S., Erdogan, H., 2012. Audio-Visual speech recognition with background music using single-channel source separation, in: IEEE Conference on Signal Processing and Communications Applications (SIU).
- Jaureguiberry, X., Leveau, P., Maller, S., Burred, J.J., 2011. Adaptation of source-specific dictionaries in non-negative matrix factorization for source separation, in: IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP).
- Jordan, M.I., Bishop, C.M., . An Introduction to graphical models. draft version, unpublished.
- Kim, J., cetin, M., Willsky, A.S., 2007. Nonparametric shape priors for active contour-based image segmentation. *Signal Processing* 87, 3021–3044.
- Lee, D.D., Seung, H.S., 2001. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems* 13, 556–562.
- Leon-Garcia, A., 1994. Probability and random processing for electrical engineering. Addison-Wesley.
- Rabiner, L., Juang, B.H., 1993. Fundamentals of speech recognition. Prentice Hall, Englewood Cliffs NJ.
- Rosti, A.V., Gales, M., 2001. Generalised linear Gaussian models. Technical Report. CUED/F-INFENG/TR.420, University of Cambridge.
- Rosti, A.V.I., Gales, M.J.F., 2004. Factor analysed hidden markov models for speech recognition. *Computer Speech and Language*, Issue 2 18, 181–200.

- Schmidt, M.N., Olsson, R.K., 2006. Single-channel speech separation using sparse non-negative matrix factorization, in: International Conference on Spoken Language Processing (INTERSPEECH).
- URL, 2009a. <http://pianosociety.com>.
- URL, 2009b. <http://www.itu.int/rec/T-REC-G.191/en>.
- Vincent, E., Gribonval, R., Fevotte, C., 2006. Performance measurement in blind audio source separation. *IEEE Transactions, Audio, speech, and language processing* 14, 1462–69.
- Virtanen, T., 2007. Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria. *IEEE Transactions on Audio, Speech, and Language Processing* 15, 1066–1074.
- Virtanen, T., Cemgil, A.T., 2009. Mixtures of gamma priors for non-negative matrix factorization based speech separation, in: International Conference on Independent Component Analysis and Blind Signal Separation.
- Virtanen, T., Cemgil, A.T., Godsill, S., 2008. Bayesian extensions to non-negative matrix factorization for audio signal modeling, in: IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP).
- Wessel, F., Schluter, R., Ney, H., 2000. Using posterior word probabilities for improved speech recognition, in: IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP).
- Wilson, K.W., Raj, B., Smaragdis, P., 2008a. Regularized non-negative matrix factorization with temporal dependencies for speech denoising, in: An-

nual Conference of the International Speech Communication Association (INTERSPEECH).

Wilson, K.W., Raj, B., Smaragdis, P., Divakaran, A., 2008b. Speech denoising using nonnegative matrix factorization with priors, in: IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP).

APPENDIX A

In this appendix, we show the MMSE estimate and the parameter Ψ learning similar to Rosti and Gales (2001), Ghahramani and Hinton (1997), and Rosti and Gales (2004). Assume we have a noisy observation \mathbf{y} as shown in the graphical model in Figure 4, which can be formulated as follows:

$$\mathbf{y} = \mathbf{x} + \mathbf{e}, \quad (47)$$

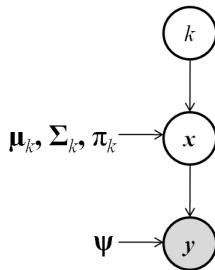


Figure 4: The graphical model of the observation model.

where \mathbf{e} is the noise term, and \mathbf{x} is the unknown underlying correct signal which needs to be estimated under a GMM prior distribution:

$$p(\mathbf{x}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad (48)$$

the error term \mathbf{e} has a Gaussian distribution with zero mean and diagonal covariance matrix Ψ :

$$p(\mathbf{e}) = \mathcal{N}(\mathbf{e}|\mathbf{0}, \Psi). \quad (49)$$

The conditional distribution of \mathbf{y} is a Gaussian with mean \mathbf{x} and diagonal covariance matrix Ψ :

$$p(\mathbf{y}|\mathbf{x}, k) = \mathcal{N}(\mathbf{y}|\mathbf{x}, \Psi). \quad (50)$$

The distribution of \mathbf{y} given the Gaussian component k is a Gaussian with mean $\boldsymbol{\mu}_k$ and diagonal covariance matrix $\boldsymbol{\Sigma}_k + \Psi$:

$$p(\mathbf{y}|k) = \mathcal{N}(\mathbf{y}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k + \Psi). \quad (51)$$

The marginal probability distribution of \mathbf{y} is a GMM:

$$p(\mathbf{y}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{y}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k + \Psi), \quad (52)$$

where the expectations $E(\mathbf{x}) = E(\mathbf{y})$, and $E(\mathbf{e}) = \mathbf{0}$. Note that, this observation model has some mathematical similarities but different concepts with factor analysis models assuming the load matrix is the identity matrix (Rosti and Gales, 2001; Ghahramani and Hinton, 1997; Rosti and Gales, 2004; Jordan and Bishop).

The MMSE estimate of \mathbf{x} can be found by calculating the conditional expectation of \mathbf{x} given the observation \mathbf{y} . Given the Gaussian component k , the joint distribution of \mathbf{x} and \mathbf{y} is a multivariate Gaussian distribution with conditional expectation and conditional variance as follows (Rosti and Gales, 2001; Leon-Garcia, 1994):

$$E(\mathbf{x}|\mathbf{y}, k) = \boldsymbol{\mu}_k + \boldsymbol{\Sigma}_k \mathbf{x} \mathbf{y} \boldsymbol{\Sigma}_k^{-1} (\mathbf{y} - \boldsymbol{\mu}_k), \quad (53)$$

$$\text{var}(\mathbf{x}|\mathbf{y}, k) = \Sigma_k - \Sigma_k \mathbf{x} \mathbf{y} \Sigma_k^{-1} \Sigma_k^T \mathbf{x} \mathbf{y}, \quad (54)$$

we know that

$$\Sigma_k \mathbf{y} = \Sigma_k + \Psi, \quad (55)$$

and

$$\begin{aligned} \Sigma_k \mathbf{x} \mathbf{y} &= \text{cov}(\mathbf{x}, \mathbf{y}) \\ &= E(\mathbf{x} \mathbf{y}^T) - E(\mathbf{x}) E(\mathbf{y}^T) \\ &= E[\mathbf{x}(\mathbf{x}^T + \mathbf{e}^T)] - E(\mathbf{x}) E(\mathbf{y}^T) \\ &= E(\mathbf{x} \mathbf{x}^T) + E(\mathbf{x}) E(\mathbf{e}^T) - E(\mathbf{x}) E(\mathbf{y}^T) \\ &= \text{var}(\mathbf{x}) + E(\mathbf{x}) E(\mathbf{x}^T) - E(\mathbf{x}) E(\mathbf{y}^T) \\ &= \text{var}(\mathbf{x}) = \Sigma_k. \end{aligned} \quad (56)$$

The conditional expectation given the Gaussian component k of the prior model is

$$\begin{aligned} E(\mathbf{x}|\mathbf{y}, k) &= \boldsymbol{\mu}_k + \Sigma_k (\Sigma_k + \Psi)^{-1} (\mathbf{y} - \boldsymbol{\mu}_k) \\ &= \hat{\mathbf{x}}_k. \end{aligned} \quad (57)$$

We also can find the following conditional expectation given only the observation \mathbf{y} as follows:

$$\begin{aligned}
E(\mathbf{x}|\mathbf{y}) &= \sum_{k=1}^K E(k|\mathbf{y}) E(\mathbf{x}|\mathbf{y}, k) \\
&= \sum_{k=1}^K \gamma_k E(\mathbf{x}|\mathbf{y}, k) \\
&= \hat{\mathbf{x}},
\end{aligned} \tag{58}$$

where

$$E(k|\mathbf{y}) = \frac{\pi_k p(\mathbf{y}|k)}{\sum_{j=1}^K \pi_j p(\mathbf{y}|j)} = \gamma_k. \tag{59}$$

From equations (57, 58, 59) we can write the final MMSE estimate of \mathbf{x} given the model parameters as follows:

$$\hat{\mathbf{x}} = \sum_{k=1}^K \gamma_k [\boldsymbol{\mu}_k + \boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1} (\mathbf{y} - \boldsymbol{\mu}_k)]. \tag{60}$$

We need also to find the following sufficient statistics to be used in estimating the model parameters:

$$\text{var}(\mathbf{x}|\mathbf{y}, k) = \boldsymbol{\Sigma}_k - \boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1} \boldsymbol{\Sigma}_k^T, \tag{61}$$

$$\begin{aligned}
E(\mathbf{x}\mathbf{x}^T|\mathbf{y}, k) &= \text{var}(\mathbf{x}|\mathbf{y}, k) + E(\mathbf{x}|\mathbf{y}, k) E(\mathbf{x}|\mathbf{y}, k)^T \\
&= \boldsymbol{\Sigma}_k - \boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1} \boldsymbol{\Sigma}_k^T + \hat{\mathbf{x}}_k \hat{\mathbf{x}}_k^T \\
&= \hat{\mathbf{R}}_k,
\end{aligned} \tag{62}$$

and

$$\begin{aligned}
E(\mathbf{x}\mathbf{x}^T|\mathbf{y}) &= \sum_{k=1}^K E(k|\mathbf{y}) E(\mathbf{x}\mathbf{x}^T|\mathbf{y}, k) \\
&= \sum_{k=1}^K \gamma_k E(\mathbf{x}\mathbf{x}^T|\mathbf{y}, k) \\
&= \sum_{k=1}^K \gamma_k \hat{\mathbf{R}}_k \\
&= \hat{\mathbf{R}}.
\end{aligned} \tag{63}$$

Parameters learning using the EM algorithm

In the training stage, we assume we have clean data with $\mathbf{e} = \mathbf{0}$. The prior GMM parameters $\pi, \boldsymbol{\mu}, \boldsymbol{\Sigma}$ are learned as regular GMM models. The only parameter that need to be estimated is $\boldsymbol{\Psi}$, which is learned from the deformed signal “ \mathbf{q}_n ” in the paper. The parameter $\boldsymbol{\Psi}$ is learned iteratively using maximum likelihood estimation. Given the data points $\mathbf{q} = \mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n, \dots, \mathbf{q}_N$, and the GMM parameters, we need to find an estimate for $\boldsymbol{\Psi}$. We follow the same procedures as in Rosti and Gales (2001), Ghahramani and Hinton (1997), and Rosti and Gales (2004).

Lets rewrite the sufficient statistics in equations (59, 57, 60, 62, 63) after replacing \mathbf{x} with \mathbf{z} (to avoid confusion between calculating MMSE and training the model parameters) as follows:

$$\gamma_{kn} = \frac{\pi_k \mathcal{N}(\mathbf{q}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k + \boldsymbol{\Psi})}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{q}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j + \boldsymbol{\Psi})}, \tag{64}$$

$$\hat{\mathbf{z}}_{kn} = E(\mathbf{z} | \mathbf{q}_n, k) = \boldsymbol{\mu}_k + \boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \boldsymbol{\Psi})^{-1} (\mathbf{q}_n - \boldsymbol{\mu}_k), \tag{65}$$

$$\hat{\mathbf{z}}_n = E(\mathbf{z}|\mathbf{q}_n) = \sum_{k=1}^K \gamma_{kn} \hat{\mathbf{z}}_{kn}, \quad (66)$$

$$\hat{\mathbf{R}}_{kn} = E(\mathbf{z}\mathbf{z}^T|\mathbf{q}_n, k) = \Sigma_k - \Sigma_k (\Sigma_k + \Psi)^{-1} \Sigma_k^T + \hat{\mathbf{z}}_{kn} \hat{\mathbf{z}}_{kn}^T, \quad (67)$$

and

$$\hat{\mathbf{R}}_n = E(\mathbf{z}\mathbf{z}^T|\mathbf{q}_n) = \sum_{k=1}^K \gamma_{kn} \hat{\mathbf{R}}_{kn}. \quad (68)$$

The complete log-likelihood can be written in a product form as follows:

$$\begin{aligned} l(\mathbf{q}, \mathbf{z}, k|\boldsymbol{\mu}, \Sigma, \pi, \Psi) &= \log \prod_{n=1}^N \prod_{k=1}^K p(k)p(\mathbf{z}|k)p(\mathbf{q}_n|\mathbf{z}, k), \\ &= \log \prod_{n=1}^N \prod_{k=1}^K [\pi_k \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_k, \Sigma_k) \mathcal{N}(\mathbf{q}_n|\mathbf{z}, \Psi)]^k, \end{aligned} \quad (69)$$

$$l(\mathbf{q}, \mathbf{z}, k|\boldsymbol{\mu}, \Sigma, \pi, \Psi) = \sum_{n=1}^N \sum_{k=1}^K k \log \pi_k + \sum_{n=1}^N \sum_{k=1}^K k \log \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_k, \Sigma_k) + \sum_{n=1}^N \sum_{k=1}^K k \log \mathcal{N}(\mathbf{q}_n|\mathbf{z}, \Psi). \quad (70)$$

The conditional expectation of the complete log likelihood, which is conditioning on the observed data \mathbf{q}_n can be written as:

$$\begin{aligned} Q &= \sum_{n=1}^N \sum_{k=1}^K E_{\mathbf{q}_n}(k|\mathbf{q}_n) \log \pi_k + \sum_{n=1}^N \sum_{k=1}^K E_{\mathbf{q}_n}(k|\mathbf{q}_n) E_{\mathbf{q}_n}(\log \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_k, \Sigma_k) | \mathbf{q}_n) \\ &\quad + \sum_{n=1}^N \sum_{k=1}^K E_{\mathbf{q}_n}(k|\mathbf{q}_n) E_{\mathbf{q}_n}(\log \mathcal{N}(\mathbf{q}_n|\mathbf{z}, \Psi) | \mathbf{q}_n), \end{aligned} \quad (71)$$

given that

$$E_{\mathbf{q}_n}(k|\mathbf{q}_n) = \frac{\pi_k \mathcal{N}(\mathbf{q}_n|\boldsymbol{\mu}_k, \Sigma_k + \Psi)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{q}_n|\boldsymbol{\mu}_j, \Sigma_j + \Psi)} = \gamma_{kn}. \quad (72)$$

We can write the complete log-likelihood as follows:

$$\begin{aligned}
Q &= \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} \log \pi_k + \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n} (\log \mathcal{N}(\mathbf{z} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) | \mathbf{q}_n) \\
&\quad + \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n} (\log \mathcal{N}(\mathbf{q}_n | \mathbf{z}, \boldsymbol{\Psi}) | \mathbf{q}_n). \tag{73}
\end{aligned}$$

For the parameter $\boldsymbol{\Psi}$, we need to maximize the third part of equation (73) with respect to $\boldsymbol{\Psi}$:

$$\begin{aligned}
Q_{\mathbf{q}_n} &= \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n} (\log \mathcal{N}(\mathbf{q}_n | \mathbf{z}, \boldsymbol{\Psi}) | \mathbf{q}_n) \\
&= \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n} \left(\log \frac{1}{(2\pi)^{\frac{d}{2}} |\boldsymbol{\Psi}|^{\frac{1}{2}}} \exp \left\{ \frac{-1}{2} (\mathbf{q}_n - \mathbf{z})^T \boldsymbol{\Psi}^{-1} (\mathbf{q}_n - \mathbf{z}) \right\} | \mathbf{q}_n, k \right), \\
&= \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n} \left(\frac{-d}{2} \log(2\pi) - \frac{1}{2} \log |\boldsymbol{\Psi}| - \frac{1}{2} (\mathbf{q}_n - \mathbf{z})^T \boldsymbol{\Psi}^{-1} (\mathbf{q}_n - \mathbf{z}) | \mathbf{q}_n, k \right), \tag{74}
\end{aligned}$$

the derivative of $Q_{\mathbf{q}_n}$ with respect to $\boldsymbol{\Psi}^{-1}$ is set to zero:

$$\frac{\partial Q_{\mathbf{q}_n}}{\partial \boldsymbol{\Psi}^{-1}} = \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n} \left(\frac{1}{2} \boldsymbol{\Psi} - \frac{1}{2} (\mathbf{q}_n - \mathbf{z}) (\mathbf{q}_n - \mathbf{z})^T | \mathbf{q}_n, k \right) = \mathbf{0}, \tag{75}$$

$$\begin{aligned}
\boldsymbol{\Psi} \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} &= \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} \mathbf{q}_n \mathbf{q}_n^T - \sum_{n=1}^N \mathbf{q}_n \sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n} (\mathbf{z} | \mathbf{q}_n, k)^T \\
&\quad - \left(\sum_{n=1}^N \mathbf{q}_n \sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n} (\mathbf{z} | \mathbf{q}_n, k)^T \right)^T + \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n} (\mathbf{z} \mathbf{z}^T | \mathbf{q}_n, k), \tag{76}
\end{aligned}$$

we know that

$$\sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} = N \quad \text{and} \quad \sum_{k=1}^K \gamma_{kn} = 1,$$

then

$$\sum_{n=1}^N \mathbf{q}_n \mathbf{q}_n^T \sum_{k=1}^K \gamma_{kn} = \sum_{n=1}^N \mathbf{q}_n \mathbf{q}_n^T,$$

and

$$\mathbf{\Psi} \sum_{n=1}^N \sum_{k=1}^K \gamma_{kn} = N \mathbf{\Psi}.$$

We can use the values of $\sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n}(\mathbf{z} | \mathbf{q}_n, k)$ and $\sum_{k=1}^K \gamma_{kn} E_{\mathbf{q}_n}(\mathbf{z} \mathbf{z}^T | \mathbf{q}_n, k)$ from equations (66, 68) to find the estimate of $\mathbf{\Psi}$ as follows:

$$\hat{\mathbf{\Psi}} = \text{diag} \left\{ \frac{1}{N} \sum_{n=1}^N \left(\mathbf{q}_n \mathbf{q}_n^T - \mathbf{q}_n \hat{\mathbf{z}}_n^T - \hat{\mathbf{z}}_n \mathbf{q}_n^T + \hat{R}_n \right) \right\}, \quad (77)$$

where the “diag” operator sets all the off-diagonal elements of a matrix to zero.

APPENDIX B

In this appendix, we show the gradients of the penalty term in the regularized NMF cost function in section 2.1. To calculate the update rule for the gains matrix \mathbf{G} , the gradients $\nabla_{\mathbf{G}}^+ L(\mathbf{G})$ and $\nabla_{\mathbf{G}}^- L(\mathbf{G})$ are needed to be calculated. Lets recall the regularized NMF cost function

$$C(\mathbf{G}) = D_{IS}(\mathbf{V} \| \mathbf{B}\mathbf{G}) + \alpha L(\mathbf{G}), \quad (78)$$

where

$$L(\mathbf{G}) = \sum_n \left\| \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} - \exp(f(\mathbf{g}_n)) \right\|_2^2, \quad (79)$$

$$f(\mathbf{g}_n) = \sum_{k=1}^K \gamma_{kn} \left[\boldsymbol{\mu}_k + \boldsymbol{\Sigma}_k (\boldsymbol{\Sigma}_k + \mathbf{\Psi})^{-1} \left(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} - \boldsymbol{\mu}_k \right) \right], \quad (80)$$

and

$$\gamma_{k_n} = \left[\frac{\pi_k \mathcal{N}\left(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k + \boldsymbol{\Psi}\right)}{\sum_{j=1}^K \pi_j \mathcal{N}\left(\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2} | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j + \boldsymbol{\Psi}\right)} \right]. \quad (81)$$

Since the training data for the GMM models are the logarithm of the normalized vectors, then the mean vectors of the GMM are always not positive, also the values of $\log \frac{\mathbf{g}_n}{\|\mathbf{g}_n\|_2}$ are also not positive, and \mathbf{g}_n is always nonnegative.

Let $\mathbf{g}_n = \mathbf{x}$, and its component a is $\mathbf{g}_{n_a} = x_a$, and $f(\mathbf{g}_n) = f(\mathbf{x})$. We can write the constraint in equation (79) as:

$$L(\mathbf{x}) = \left\| \frac{\mathbf{x}}{\|\mathbf{x}\|_2} - \exp(f(\mathbf{x})) \right\|_2^2. \quad (82)$$

The a component of the gradient of $L(\mathbf{x})$ is

$$\begin{aligned} \frac{\partial L(\mathbf{x})}{\partial x_a} &= 2 \left(\frac{x_a}{\|\mathbf{x}\|_2} - \exp(f(x_a)) \right) \left(\frac{1}{\|\mathbf{x}\|_2} - \frac{x_a^2}{\|\mathbf{x}\|_2^3} - \nabla f(x_a) \exp(f(x_a)) \right) \\ &= \nabla L(x_a), \end{aligned} \quad (83)$$

which can be written as a difference of two positive terms

$$\nabla L(x_a) = \nabla^+ L(x_a) - \nabla^- L(x_a). \quad (84)$$

The component a of the gradient of $f(\mathbf{x})$ can be written as a difference of two positive terms:

$$\frac{\partial f(\mathbf{x})}{\partial x_a} = \nabla^+ f(x_a) - \nabla^- f(x_a). \quad (85)$$

The component a of the gradient of $L(\mathbf{x})$ in equation (84) can be written as:

$$\nabla^+ L(x_a) = 2 \left\{ \frac{x_a}{\|\mathbf{x}\|_2} \left(\frac{1}{\|\mathbf{x}\|_2} + \exp(f(x_a)) \nabla^- f(x_a) \right) + \exp(f(x_a)) \left(\frac{x_a^2}{\|\mathbf{x}\|_2^3} + \exp(f(x_a)) \nabla^+ f(x_a) \right) \right\}, \quad (86)$$

and

$$\nabla^- L(x_a) = 2 \left\{ \frac{x_a}{\|\mathbf{x}\|_2} \left(\frac{x_a^2}{\|\mathbf{x}\|_2^3} + \exp(f(x_a)) \nabla^+ f(x_a) \right) + \exp(f(x_a)) \left(\frac{1}{\|\mathbf{x}\|_2} + \exp(f(x_a)) \nabla^- f(x_a) \right) \right\}. \quad (87)$$

We need to find the values of $\nabla^+ f(x_a)$ and $\nabla^- f(x_a)$. Note that, the term $\Sigma_k (\Sigma_k + \Psi)^{-1}$ forms a diagonal matrix.

Let

$$H(x_a) = \boldsymbol{\mu}_{k_a} + \Sigma_{k_{aa}} (\Sigma_{k_{aa}} + \Psi_{aa})^{-1} \left(\log \frac{x_a}{\|\mathbf{x}\|_2} - \boldsymbol{\mu}_{k_a} \right), \quad (88)$$

then $f(\mathbf{x})$ in equation (80) can be written as:

$$f(\mathbf{x}) = \sum_{k=1}^K \gamma_k(\mathbf{x}) H(x_a). \quad (89)$$

The gradient of $f(\mathbf{x})$ in equation (89) can be written as:

$$\nabla f(x_a) = \sum_{k=1}^K [\gamma_k(\mathbf{x}) \nabla H(x_a) + H(x_a) \nabla \gamma_k(x_a)], \quad (90)$$

where

$$\gamma_k(\mathbf{x}) = \left[\frac{\pi_k \mathcal{N} \left(\log \frac{\mathbf{x}}{\|\mathbf{x}\|_2} | \boldsymbol{\mu}_k, \Sigma_k + \Psi \right)}{\sum_{j=1}^K \pi_j \mathcal{N} \left(\log \frac{\mathbf{x}}{\|\mathbf{x}\|_2} | \boldsymbol{\mu}_j, \Sigma_j + \Psi \right)} \right] = \frac{M_k(\mathbf{x})}{N_k(\mathbf{x})}. \quad (91)$$

We can also write the gradient components of $H(x_a)$ and $\gamma_k(\mathbf{x})$ as a difference of two positive terms

$$\nabla H(x_a) = \nabla^+ H(x_a) - \nabla^- H(x_a), \quad (92)$$

and

$$\nabla \gamma_k(x_a) = \nabla^+ \gamma_k(x_a) - \nabla^- \gamma_k(x_a). \quad (93)$$

The gradient of $f(x_a)$ in equations (85, 90) can be written as:

$$\nabla^+ f(x_a) = \sum_{k=1}^K [\gamma_k(\mathbf{x}) \nabla^+ H(x_a) + H^+(x_a) \nabla^+ \gamma_k(x_a) + H^-(x_a) \nabla^- \gamma_k(x_a)], \quad (94)$$

$$\nabla^- f(x_a) = \sum_{k=1}^K [\gamma_k(\mathbf{x}) \nabla^- H(x_a) + H^-(x_a) \nabla^+ \gamma_k(x_a) + H^+(x_a) \nabla^- \gamma_k(x_a)], \quad (95)$$

where

$$\nabla^+ H(x_a) = \Sigma_{k_{aa}} (\Sigma_{k_{aa}} + \Psi_{aa})^{-1} \frac{1}{x_a}, \quad (96)$$

$$\nabla^- H(x_a) = \Sigma_{k_{aa}} (\Sigma_{k_{aa}} + \Psi_{aa})^{-1} \frac{x_a}{\|\mathbf{x}\|_2^2}, \quad (97)$$

and $H(x_a)$ can be written as a difference of two positive terms:

$$H(x_a) = H^+(x_a) - H^-(x_a), \quad (98)$$

where

$$H^+(x_a) = -\Sigma_{k_{aa}} (\Sigma_{k_{aa}} + \Psi_{aa})^{-1} \boldsymbol{\mu}_{k_a}, \quad (99)$$

and

$$H^-(x_a) = -\left[\boldsymbol{\mu}_{k_a} + \Sigma_{k_{aa}} (\Sigma_{k_{aa}} + \Psi_{aa})^{-1} \log \frac{x_a}{\|\mathbf{x}\|_2} \right]. \quad (100)$$

We can rewrite $\gamma_k(\mathbf{x})$ in equation (91) as:

$$\gamma_k(\mathbf{x}) = \frac{M_k(\mathbf{x})}{N_k(\mathbf{x})}, \quad (101)$$

note that $\gamma_k(\mathbf{x}), M_k(\mathbf{x}), N_k(\mathbf{x}) \geq 0$.

The component a of the gradient of $\gamma_k(\mathbf{x})$ can be written as:

$$\nabla \gamma_k(x_a) = \frac{N_k(\mathbf{x}) \nabla M_k(x_a) - M_k(\mathbf{x}) \nabla N_k(x_a)}{N_k^2(\mathbf{x})}. \quad (102)$$

We can write the gradients of $M_k(\mathbf{x})$ and $N_k(\mathbf{x})$ as a difference of two positive terms

$$\nabla M_k(x_a) = \nabla^+ M_k(x_a) - \nabla^- M_k(x_a), \quad (103)$$

and

$$\nabla N_k(x_a) = \sum_{k=1}^K \nabla^+ M_k(x_a) - \sum_{k=1}^K \nabla^- M_k(x_a). \quad (104)$$

The gradient of $\gamma_k(x_a)$ in equation (93) can be written as:

$$\nabla^+ \gamma_k(x_a) = \frac{N_k(\mathbf{x}) \nabla M_k^+(x_a) + M_k(\mathbf{x}) \sum_{k=1}^K \nabla^- M_k(x_a)}{N_k^2(\mathbf{x})}, \quad (105)$$

$$\nabla^- \gamma_k(x_a) = \frac{N_k(\mathbf{x}) \nabla M_k^-(x_a) + M_k(\mathbf{x}) \sum_{k=1}^K \nabla^+ M_k(x_a)}{N_k^2(\mathbf{x})}, \quad (106)$$

where

$$\nabla^+ M_k(x_a) = M_k(\mathbf{x}) (\Sigma_{k_{aa}} + \Psi_{aa})^{-1} \left[\frac{-1}{x_a} \log \frac{x_a}{\|\mathbf{x}\|_2} - \frac{\boldsymbol{\mu}_{k_a} x_a}{\|\mathbf{x}\|_2^2} \right], \quad (107)$$

and

$$\nabla^- M_k(x_a) = M_k(\mathbf{x}) (\Sigma_{k_{aa}} + \Psi_{aa})^{-1} \left[\frac{-\boldsymbol{\mu}_{k_a}}{x_a} - \frac{x_a}{\|\mathbf{x}\|_2^2} \log \frac{x_a}{\|\mathbf{x}\|_2} \right]. \quad (108)$$

After finding $\nabla^+ \gamma_k(x_a)$, and $\nabla^- \gamma_k(x_a)$ from equations (105, 106), and $\nabla^+ H(x_a)$, and $\nabla^- H(x_a)$ from equations (96, 97), we can find the gradients $\nabla^+ f(x_a)$, and $\nabla^- f(x_a)$ in equations (94, 95), which complete our solution for $\nabla^+ L(x_a)$, and $\nabla^- L(x_a)$ in equations (86, 87).